UNITED STATES PATENT APPLICATION FOR:

METHOD, ARTICLE OF MANUFACTURE AND APPARATUS FOR PROCESSING REDUNDANT ELECTRONIC MAIL MESSAGES

INVENTORS:

CARY LEE BATES

PAUL REUBEN DAY

JOHN MATTHEW SANTOSUOSSO

Certification Under 37 CFR 1.10

I hereby certify that this New Application and the documents referred to as enclosed therein are being deposited with the United States Postal Service on <u>June 13, 2000</u>, in an envelope marked as "Express Mail United States Postal Service," Mailing Label No. <u>EL635441747US</u> to: Assistant Commissioner for Patents, Box PATENT APPLICATION, Washington, D.C. 20231.

B. Todd Patterson
Name

B. Todd Patterson
Signature

_6/13/2000_
Date of Signature

METHOD, ARTICLE OF MANUFACTURE AND APPARATUS
FOR PROCESSING REDUNDANT ELECTRONIC MAIL MESSAGES

5   ## BACKGROUND OF THE INVENTION

Field of the Invention

The invention relates to electronic mail systems and, more particularly, the invention relates to a method, article of manufacture and apparatus for processing redundant electronic mail messages.

10

Background of the Related Art

In today's electronic mail (e-mail) environment, many users or recipients of e-mail messages often forward the contents of received e-mail message to other e-mail users. The recipient may append additional comments to the received e-mail message

15   or simply forward the received e-mail message to other e-mail users.

One problem associated with the utilization of forwarded messages is that a particular user or recipient may repeatedly receive the same e-mail message from different senders. The recipient may eventually become cluttered with redundant e-mail messages or become annoyed with reading or receiving these redundant e-mail

20   messages.

Therefore, a need exists for a method, article of manufacture and apparatus for processing redundant e-mail messages.

SUMMARY OF THE INVENTION

25   The invention provides a method, article of manufacture and apparatus for processing redundant electronic mail (e-mail) messages. In one embodiment, the invention provides a method for processing an e-mail message comprising receiving the e-mail message, and deleting the received e-mail message if the received e-mail message is substantially similar to an available e-mail message. As such, the received

30   e-mail message is deleted if the received e-mail message is considered redundant to a recipient terminal.

In another embodiment, the invention provides a method for processing redundant e-mail messages comprising forwarding an e-mail message addressed to at least one recipient terminal, determining whether the forwarded e-mail message is substantially similar to an available e-mail message, and deleting the forwarded e-mail

5 message if the forwarded e-mail message is substantially similar to the available e-mail message.

The invention also provides an apparatus comprising a memory and a processing unit, where the apparatus is configured for processing redundant e-mail messages. The memory stores an e-mail program. The processing unit executes the e-

10 mail program retrieved from the memory, receives an e-mail message, and deletes the received e-mail message if the received e-mail message is substantially similar to an available e-mail message.

Additionally, a computer readable medium storing a software program is provided, where the software program, when executed by a computer, causes the

15 computer to perform a method. In one embodiment of the computer readable medium, the method comprises receiving an electronic mail message, and deleting the received electronic mail message if the received electronic mail message is substantially similar to an available electronic mail message.

20 **BRIEF DESCRIPTION OF THE DRAWINGS**

The teachings of the present invention can be readily understood by considering the following detailed description in conjunction with the accompanying drawings, in which:

FIG. 1 depicts a simplified block diagram of an electronic mail (e-mail) system

25 that benefits from the present invention;

FIG. 2 depicts a simplified block diagram of a user terminal (computer) within the electronic mail system of FIG. 1;

FIG. 3 depicts a flow diagram of a mail program for processing e-mail messages in accordance with the present invention;

30 FIG. 4 depicts a flow diagram of a mail program for handling an opened message; and

FIG. 5 depicts a flow diagram of the mail program for configuring e-mail

messages in accordance with the present invention.

To facilitate understanding, identical reference numerals have been used, where possible, to designate identical elements that are common to the figures.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

5

FIG. 1 depicts a block diagram of an electronic mail (e-mail) system 100 that benefits from the use of the present invention. One embodiment of the present invention comprises an executable software routine(s) that generally resides within the user terminals 106. The system 100 comprises an e-mail server 102, a network 104 and

10 a plurality of user terminals 106_1, 106_2,...106_n. Each user terminal may be a computer system, an e-mail enabled network appliance, an e-mail enabled wireless device and the like. The user terminals 106 are connected to one another through the network 104 and also connected through the network 104 to the e-mail server 102. The e-mail server 102 provides e-mail services to all of the user terminals 106 such that e-mail addressed

15 from one terminal to another is routed through the network 104 to the e-mail server 102. Each user terminal 106 may then retrieve its e-mail from the e-mail server 102. Although the system is shown having a hub and spoke configuration, other forms of networks may also benefit from the invention.

FIG. 2 depicts a functional block diagram of a user terminal 106, e.g., a general

20 purpose computer system. User terminal 106 comprises a computer 200 and an input/output device or devices 202 such as a monitor, a keyboard, a mouse, a trackball and the like. The computer 200 comprises a central processing unit 204, support components 206, input/output circuits 208 and a memory 210. The memory 210 may be a solid state memory, a disc drive, an optical memory, a removable memory device,

25 or a combination of any of these memory devices. The input/output circuits 208 provide a well-known interface from the CPU 204 to the I/O devices 202. The support components 206 are well known in the art and include such circuits as cache, power supplies, clock circuits, and the like. The combination of all of these circuits, components and elements forms a general purpose computer that, when executing a

30 particular software package or routine, is a specific purpose computer. In this case, the CPU 204, when executing the electronic mail processing software 212 of the present invention, operates as an e-mail processor 216. The e-mail software 212 resides in

memory 210. In addition, the e-mail messages are stored in a message storage 214 within the memory 210. In addition, the I/O circuits 208 generally contain a network interface card or modem that couples the computer to the network of FIG. 1 such that e-mail messages can be routed to and from the computer 200.

5   In one embodiment of the present invention, the e-mail processor 216 forwards an e-mail message to a recipient terminal. Upon processing the forwarded e-mail message, the recipient terminal deletes the forwarded e-mail message if the recipient terminal identifies an available e-mail message that is substantially similar to the forwarded e-mail message. As such, the forwarded e-mail message is deleted if the e-

10   mail message is considered redundant to the recipient terminal.

As will be described in detail below, another embodiment of the present invention is implemented as a program product for use with a computer system such as, for example, the system 100 shown in FIG. 1. The program(s) of the program product defines functions of the preferred embodiment and can be contained on a variety of

15   signal/bearing media, which include, but are not limited to, (i) information permanently stored on non-writable storage media, (e.g., read-only memory devices within a computer such as CD-ROM disks readable by a CD-ROM drive); (ii) alterable information stored on writable storage media (e.g., floppy disks within a diskette drive or hard-disk drive); or (iii) information conveyed to a computer by a communications

20   medium, such as through a computer or telephone network, including wireless communications. Such signal-bearing media, when carrying computer-readable instructions that direct the functions of the present invention, represent embodiments of the present invention.

FIG. 3 depicts a flow diagram of a mail program for processing e-mail messages

25   in accordance with the present invention. FIG. 4 depicts a flow diagram of a mail program for handling an opened message. FIG. 5 depicts a flow diagram of the mail program for configuring an e-mail message in accordance with the present invention. To best understand the invention, the reader should simultaneously refer to FIGS. 3-5.

Referring to FIG. 3, the mail program is embodied in a method 300 that starts at

30   step 302 and proceeds to retrieve an event at step 304. Such events may represent a particular command from a user, e.g., opening an e-mail message, retrieving incoming e-mail messages, or confirming the deletion of marked e-mail messages. The method

5

300 operates in an event-driven manner, i.e., the method 300 retrieves events and performs different steps depending on a particular event retrieved.

At step 306, the method 300 queries whether the event is to open an e-mail message. If the event is to open an e-mail message, the method 300 proceeds to step 308, where an open message task is started. The open message task is embodied in a method 400 as described below with respect to FIG. 4. This open message task may be processed independently of method 300. After starting the open message task, the method 300 retrieves the next event at step 304.

If the event is not to open an e-mail message, the method 300 proceeds to step 310, where the query is whether the event is to retrieve incoming e-mail messages, i.e., get e-mail messages from the server 102. If the event is to retrieve incoming e-mail messages, the method 300 proceeds to reset the event timer at step 312 and query whether there are incoming e-mail message to retrieve at step 314. If there are no incoming e-mail messages to retrieve or no additional incoming e-mail messages to retrieve, the method 300 proceeds to retrieve the next event at step 304. If there are more incoming e-mail messages to retrieve, the method 300 proceeds to step 316, where an incoming e-mail message is retrieved from the e-mail server 102.

At step 318, the method 300 queries whether the incoming e-mail message is flagged in accordance with the present invention, i.e., if the e-mail message is flagged, the recipient terminal determines whether to delete the incoming e-mail message. Control information embedded in the e-mail message, e.g., in a header, may be used to flag the e-mail message.

In one embodiment of the invention, a sender of the incoming e-mail message may flag or designate the e-mail message as an e-mail message that may be deleted if redundant. This embodiment is further described below in FIG. 5. In another embodiment of the invention, the recipient of the incoming e-mail message may also flag this e-mail message as an e-mail message that may be deleted if redundant. In this case, the recipient may flag the incoming e-mail message if the incoming or received e-mail message was substantially unaltered by a sender forwarding the e-mail message. In other words, the recipient may flag the incoming e-mail message if the sender forwarding an e-mail message did not provide additional text or attachments.

If the incoming e-mail message is not flagged, the method 300 returns to step

6

314 to determine whether additional e-mal messages are queued for retrieval. If the incoming e-mail message is flagged, the method proceeds to step 320, where a query is performed to determine whether the flagged e-mail message is at least substantially similar to an available e-mail message in the message storage 214. Accordingly, the

5   method 300 determines whether to delete the incoming e-mail message only if the e-mail message is flagged.

The available e-mail messages contained in the message storage 214 comprise previously received, undeleted e-mail messages, which may include previously opened e-mail messages and unopened e-mail messages. Moreover, an available e-mail

10   message contained in the message storage 214 is substantially similar to the flagged e-mail message if the identifier field of the available e-mail message is substantially similar to the identifier field of the flagged e-mail message. Information in the identifier field may comprise, for example, user information, subject information and timestamp information, e.g., date and time of a sent e-mail message.

15   If no available e-mail messages exist in the message storage 214, the method 300 proceeds to determine whether additional incoming e-mail messages are to be retrieved at step 314. If an available e-mail message exists in the message storage 214, the method 300 proceeds to step 322, where a query determines whether to automatically delete the flagged e-mail messages. Namely, step 322 determines

20   whether a user has selected an automatic delete mode. If the automatic delete mode has been selected, the method 300 proceeds to delete the incoming e-mail message at step 324 and to determine whether more incoming e-mail messages are to be retrieved at step 314. If the automatic delete mode has not been selected, the method 300 proceeds to step 326, where the incoming e-mail message is marked as a duplicate message.

25   After marking the e-mail message, the method 300 proceeds to determine whether additional incoming e-mail messages are to be retrieved at step 314. Steps 314 through 326 are repeated for each incoming e-mail message that is retrieved.

Returning to step 310, if the event is not to retrieve incoming e-mail messages, the method 300 proceeds to step 328, where a query determines whether the event is to

30   delete marked e-mail messages. In the context of the present invention, such marked e-mail messages comprise duplicate or redundant e-mail messages. If the event is to delete marked e-mail messages, the method 300 proceeds to delete all marked e-mail

messages at step 330 and proceeds to retrieve the next event at step 304. If the event is not to delete marked e-mail messages, the method 300 proceeds to step 332.

At step 332, the method 300 performs a query to determine whether the event is to edit a mailing list or a distribution list. Such a mailing list is utilized to conveniently send, e.g., forward, an e-mail message to at least one recipient or user. If the event is to edit the mailing list, the method 300 proceeds to step 334, where the mailing list is opened. Once the mailing list is opened, step 334 also enables a user to edit the mailing list and to specify sending attributes, e.g., attributes or features to implement when sending e-mail messages. A recipient terminal would process these attributes in substantially the same manner as the flag described with respect to step 318. In other words, the recipient of the incoming e-mail message may perform specific functions upon processing the attribute for the mailing list. For example, the recipient may determine whether to delete an incoming e-mail message if the attribute is configured for the recipient to delete redundant e-mail messages. Other attributes may specify the e-mail message as URGENT, FYI (for your information) or Automatic Acknowledgement. Moreover, certain attributes may be applied to specific recipients on the mailing list.

After opening the mailing list and specifying attributes, the method 300 proceeds to retrieve the next event at step 304. If the event is not to edit the mailing list, the method 300 proceeds to handle all other events at step 336 before retrieving the next event at step 304.

Referring to FIG. 4, the method 400 embodies an open message task that may be processed concurrently with the method 300 of the e-mail program. Method 400 starts at step 402 and proceeds to step 404 where a window is opened and the open message is displayed in the open window. The method 400 proceeds to retrieve an event at step 406.

At step 408, the method 400 determines whether the event is to send, e.g., forward or reply to, an e-mail message in response to the opened e-mail message. If the event is to send an e-mail message, the method 400 proceeds to step 412, where a send message task is started or activated. The send message task is embodied in a method 500 as described below in FIG. 5. After starting the send task, the method 400 proceeds to retrieve the next event at step 406. If the event is not to send an e-mail

message, the method 400 proceeds to handle all other events in a normal manner in step 410 before retrieving the next event at step 406.

Referring to FIG. 5, the method 500 embodies a send task for configuring e-mail messages in accordance with the present invention. Method 500 starts at step 502 and proceeds to open an edit window at step 504. The open edit window displays the e-mail message to be sent. The method 500 proceeds to retrieve an event at step 506.

At step 508, the method 500 queries whether the event is to send an e-mail message in accordance with the present invention, e.g., send an e-mail message that is not redundant to a recipient terminal. If the event is to send a forward message in this manner, the method 500 proceeds to step 510 where the e-mail message to be sent is flagged. This flag may be provided in a control word embedded in the e-mail message. The method 500 proceeds to send the e-mail message at step 512, close the open edit window at step 514, and exit the method 500 at step 516.

In response to an e-mail message flagged and sent in respective steps 510 and 512, the recipient terminal processes the flag to determine whether to delete the sent, e.g., forwarded, e-mail message. This step was previously discussed with respect to step 318. The recipient terminal deletes the forwarded e-mail message if the forwarded e-mail message is substantially similar to an available e-mail message at step 320.

Returning to step 508, if the event is not to send a message that is not redundant to the recipient terminal, the method 500 proceeds to step 518, where a query determines whether the event is some other form of sending an e-mail message. If the event is not some other form of sending an e-mail message, the method 500 proceeds to perform normal send e-mail processing at step 520 and proceeds to retrieve the next event at step 506. If the event is some other form of sending an e-mail message, the method 500 proceeds to step 522 where a query determines whether an e-mail message is being sent to recipients addressed on a mailing list, e.g., a named distribution list. Namely, step 522 determines whether the e-mail message to be sent is linked to a distribution list.

If the e-mail message is not to be sent to a mailing list, the method 500 proceeds to send the e-mail message at step 524, close the edit window at step 526, and exit the method 500 at step 528. If the e-mail message is to be sent to a mailing list, the method 500 proceeds to step 530, where any "send attributes" are retrieved for the mailing list.

These attributes may be configured in control information, e.g., a control word, sent in a header of an e-mail message. The attributes may be stored and retrieved with the mailing list. Different types of attributes are described above with respect to step 334.

At step 532, the method 500 queries whether an attribute is configured to provide a forwarded e-mail message that is not redundant to the recipient, i.e., a "non-redundant" attribute. If the non-redundant attribute is detected, the method 500 proceeds to step 534, where the e-mail message to be sent is flagged as a non-redundant forward message. The method 500 proceeds to step 536. If the non-redundant attribute is not detected, the method 500 directly proceeds to step 536.

At step 536, the method 500 queries whether a FYI (for your information) attribute has been configured. The FYI attribute is used when an e-mail message to be sent is of low importance. If the FYI attribute has been detected, the method 500 flags the message as a FYI message at step 538 and sends the e-mail message at step 540. If the FYI attribute is not detected, the method 500 directly proceeds to send the e-mail message at step 540. After step 540, the method proceeds to close the edit window at step 542, and exit the method 500 at step 544.

The method 500 may also incorporate other attributes contemplated within the scope of the present invention. For example, the method 500 may designate important e-mail messages with an URGENT attribute. If a sender desires automatic acknowledgement of a received e-mail message, then the method 500 would implement an Automatic Acknowledgement attribute.

Although various embodiments which incorporate the teachings of the present invention have been shown and described in detail herein, those skilled in the art can readily devise many other varied embodiments that still incorporate these teachings.